

<https://helda.helsinki.fi>

---

## DeepMIB : User-friendly and open-source software for training of deep learning network for biological image segmentation

Belevich, Ilya

2021-03

---

Belevich , I & Jokitalo , E 2021 , ' DeepMIB : User-friendly and open-source software for training of deep learning network for biological image segmentation ' , PLoS Computational Biology , vol. 17 , no. 3 , 1008374 . <https://doi.org/10.1371/journal.pcbi.1008374>

---

<http://hdl.handle.net/10138/328978>

<https://doi.org/10.1371/journal.pcbi.1008374>

---

cc\_by

publishedVersion

---

*Downloaded from Helda, University of Helsinki institutional repository.*

*This is an electronic reprint of the original article.*

*This reprint may differ from the original in pagination and typographic detail.*

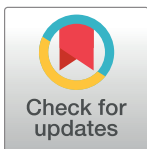
*Please cite the original version.*

## RESEARCH ARTICLE

# DeepMIB: User-friendly and open-source software for training of deep learning network for biological image segmentation

Ilya Belevich \*, Eija Jokitalo \*

Institute of Biotechnology, Helsinki Institute of Life Science, University of Helsinki, Helsinki, Finland

\* [ilya.belevich@helsinki.fi](mailto:ilya.belevich@helsinki.fi) (IB); [eija.jokitalo@helsinki.fi](mailto:eija.jokitalo@helsinki.fi) (EJ)


## Abstract

We present DeepMIB, a new software package that is capable of training convolutional neural networks for segmentation of multidimensional microscopy datasets on any workstation. We demonstrate its successful application for segmentation of 2D and 3D electron and multicolor light microscopy datasets with isotropic and anisotropic voxels. We distribute DeepMIB as both an open-source multi-platform Matlab code and as compiled standalone application for Windows, MacOS and Linux. It comes in a single package that is simple to install and use as it does not require knowledge of programming. DeepMIB is suitable for everyone interested of bringing a power of deep learning into own image segmentation workflows.

## OPEN ACCESS

**Citation:** Belevich I, Jokitalo E (2021) DeepMIB: User-friendly and open-source software for training of deep learning network for biological image segmentation. PLoS Comput Biol 17(3): e1008374. <https://doi.org/10.1371/journal.pcbi.1008374>

**Editor:** Dina Schneidman-Duhovny, Hebrew University of Jerusalem, ISRAEL

**Received:** November 5, 2020

**Accepted:** January 24, 2021

**Published:** March 2, 2021

**Copyright:** © 2021 Belevich, Jokitalo. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** All relevant data are within the manuscript and its [Supporting Information](#) files.

**Funding:** The research was supported by Biocenter Finland and Academy of Finland (projects 1287975 and 1331998). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing interests:** The authors have declared that no competing interests exist.

## Author summary

Deep learning approaches are highly sought after solutions for coping with large amounts of collected datasets and are expected to become an essential part of imaging workflows. However, in most cases, deep learning is still considered as a complex task that only image analysis experts can master. With DeepMIB we address this problem and provide the community with a user-friendly and open-source tool to train convolutional neural networks and apply them to segment 2D and 3D grayscale or multi-color datasets.

This is a *PLOS Computational Biology* Software paper.

## Introduction

During recent years, improved availability of high-performance computing resources, and especially graphics processing units (GPUs), has boosted applications of deep learning techniques into many aspects of our lives. The biological imaging is not an exception, and methods based on deep learning techniques [1] are continually emerging to deal with various tasks such as image classification [2,3], restoration [4], segmentation [5–8], and tracking [9]. Unfortunately, in many cases, the application of these methods is not easy and requires significant

knowledge in computer sciences, making it difficult to adapt by many researchers. Software developers have already started to address this challenge by developing user-friendly deep learning tools, such as Cell Profiler [10], Ilastik [11], ImageJ plug-ins DeepImageJ [12] and U-net [13], CDeep3M [5], and Uni-EM [14] that are especially suitable for biological projects. However, the overall usability is limited because they either rely on pre-trained networks without the possibility of training on new data [10–12], are limited to electron microscopy (EM) datasets [14], or have specialized computing requirements [5,13].

In our opinion, in addition to providing good results, ideal deep learning solution should fulfill the following criteria: a) is capable of training on new data; b) has a user-friendly interface; c) is easy to install and would work straight out of the box; d) is free of charge and e) has open-source code for future development. Preferably, it also would be compatible with 2D and 3D, EM and light microscopy (LM) datasets. To address all these points, we are presenting DeepMIB as a free open-source software tool for image segmentation using deep learning. DeepMIB can be used to train 2D and 3D convolutional neural networks (CNN) on user's isotropic and anisotropic EM or LM multicolor datasets. DeepMIB comes bundled with Microscopy Image Browser (MIB) [15], forming a powerful combination to address all aspects of an imaging pipeline starting from basic processing of images (*e.g.*, filtering, normalization, alignment) to manual, semi-automatic and fully automatic segmentation, proofreading of segmentations, their quantitation and visualization.

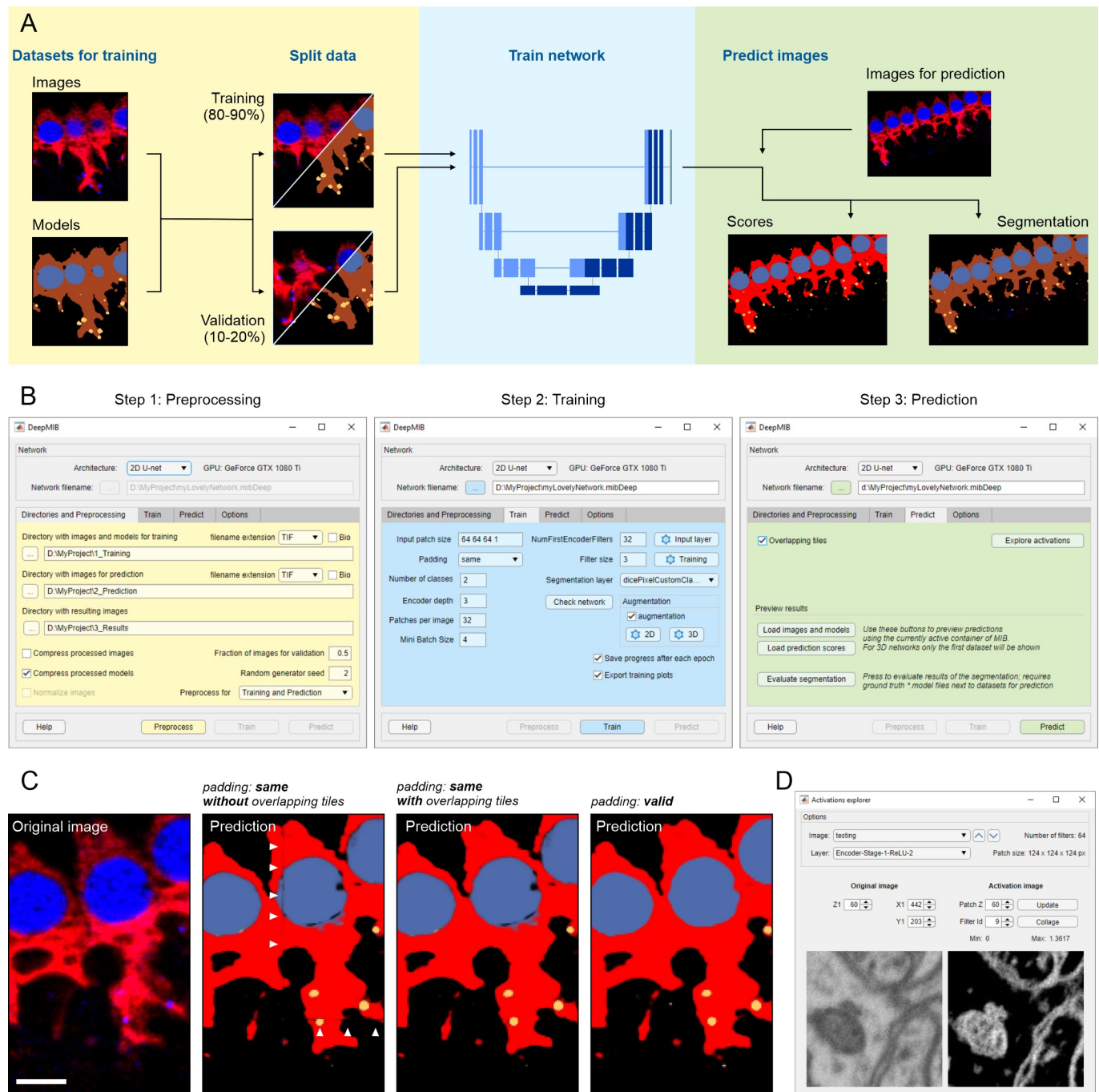
## Design and implementation

### Deep learning workflow

DeepMIB deep learning workflow comprises three main steps: preprocessing, training and prediction (Fig 1A). Preprocessing requires images supplemented with the ground truth, which can be generated directly in MIB or using external tools [10,11,16,17]. Typically, the application of deep learning to image segmentation requires large training sets. However, DeepMIB utilizes sets of 2D and 3D CNN architectures (U-Net [6,8], SegNet [18]) that can provide good results already with only few training datasets (starting from 2 to 10 images [13]), making it useful for small-scale projects too. To prevent overfitting of the network during training, the segmented images are randomly split into two sets, where the larger set (80–90%) is used for training and the smaller set for validation of the training process. DeepMIB is splitting the images automatically using configurable settings. When only one 3D dataset is available, it can be split into multiple subvolumes using the Chopping tool of MIB to dedicate one subvolume for validation. Alternatively, training can be done without validation. A modern GPU is essential for efficient training, but small datasets can still be trained using a central processing unit (CPU) only. Finally, in the prediction step, the trained networks are used to process unannotated datasets to generate prediction maps and segmentations.

### User interface and training process

The user interface of DeepMIB (available from MIB *Menu->Tools->Deep learning segmentation*) consists of a single window with a top panel and several tabs arranged in a pipeline order and color-coded to help navigation (Fig 1B). The process starts from choosing the network architecture from four options, 2D or 3D U-Net [6,8], 3D U-Net designed for anisotropic datasets (S1 Fig) or 2D SegNet [18], as preprocessing of datasets is determined according to the selected CNN. To improve usability and minimize data conversion, DeepMIB accepts both standard (*e.g.*, TIF, PNG) and microscopy (Bio-Formats [19]) image formats. The network training is done using the Train tab. DeepMIB automatically generates a network layout based on the most critical parameters that the user has to specify: the input patch size, whether to use



**Fig 1. Schematic presentation of DeepMIB deep learning workflow.** (A) A scheme of a typical deep learning workflow. (B) Stepwise workflow of DeepMIB user interface for training or/and prediction. (C) Comparison of different padding types and overlapping strategies during prediction. Description for used images is given in Fig 1D. Arrowheads denote artefacts that may appear during prediction of datasets trained with the “same” padding parameter. (D) A snapshot of the GUI of the Activations explorer tool used for visualization of network features. Scale bar, 5  $\mu$ m.

<https://doi.org/10.1371/journal.pcbi.1008374.g001>

convolutional padding, the number of classes and the depth of the network. With these parameters, the network layout can be tuned for specific datasets and available computational power. To improve end-results, data augmentation is a powerful method to expand the training set by using various transformations such as reflection, rotation, scaling, and shear. DeepMIB has

separate configurable augmentation options for 2D and 3D networks. The training set can also be extended by filtering training images and their corresponding ground truth models using Elastic Distortion [20] filter of MIB (available from MIB Menu->Image->Image filters). For final fine-tuning, the normalization settings of the input layer and thorough tweaking of training parameters can be done. It is possible to store network checkpoints after each iteration and continue training from any of those steps.

### Prediction of new datasets

The prediction process is rather simple and requires only loading of a network file and preprocessing of images for prediction. Overlapping tiles option can be used to reduce edge artefacts during prediction (Fig 1C). The results can be instantly checked in MIB and their quality can be evaluated against the ground truth with various metrics (e.g., Jaccard similarity coefficient, F1 Score), providing that the ground truth model for the prediction dataset is available. In addition, Activations Explorer provides means to follow image perturbations inside the network and explore the network features (Fig 1D).

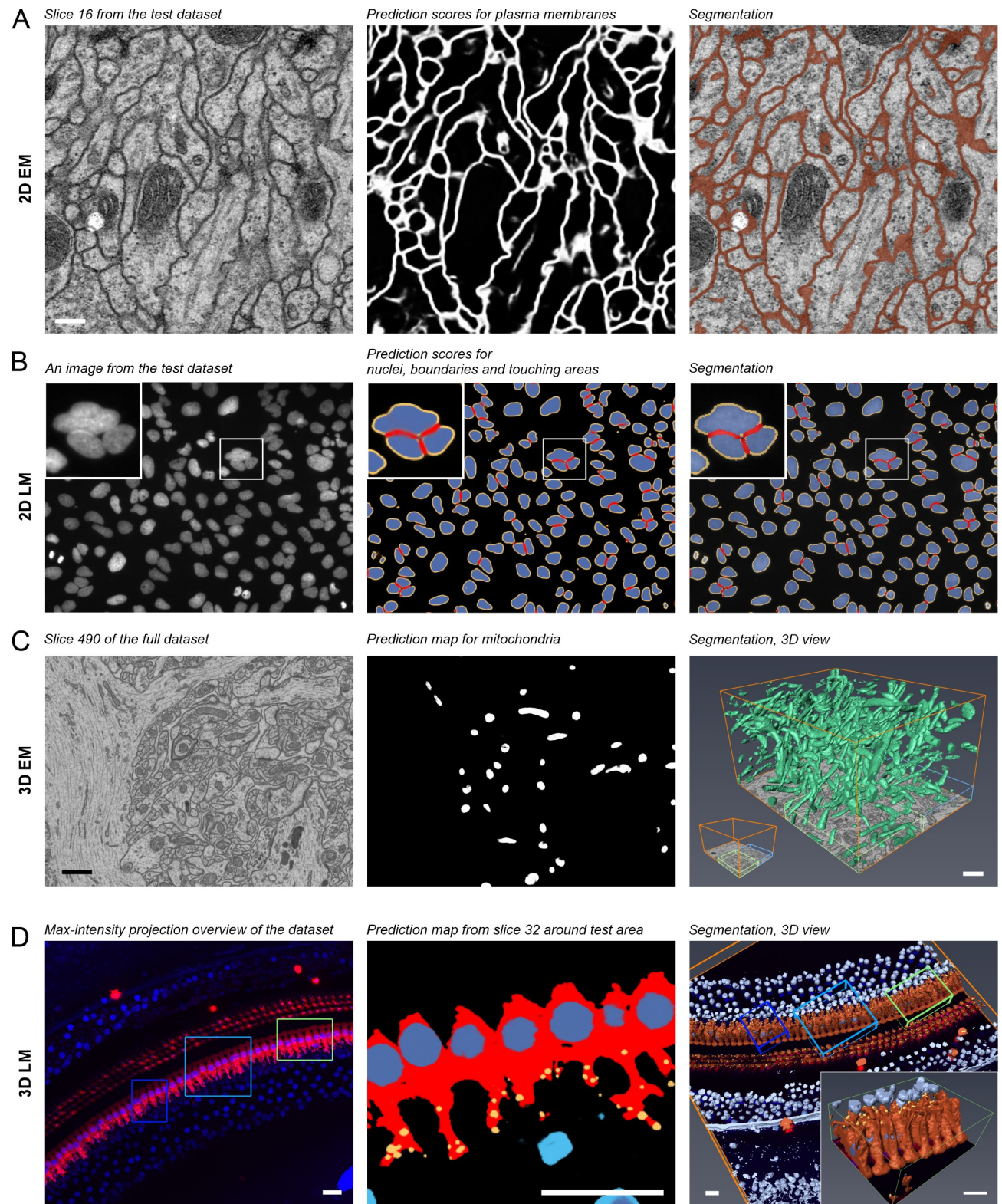
## Results

We have tested DeepMIB on several 2D and 3D datasets from both LM and EM, and here present examples from each of the four cases (Fig 2, S1–S4 Movies). These results should not be considered as winners of a segmentation challenge but rather as examples of what inexperienced users can achieve on their own with only basic knowledge of CNNs. The first example demonstrates the segmentation of membranes from a 2D-EM image, which can be individual micrographs of thin TEM sections or slices of 3D-datasets. Here, the plasma membrane from a slice of a serial section TEM dataset of first instar larva ventral nerve cord [21] of the *Drosophila melanogaster* was segmented (Fig 2A, S1 Movie). The second example demonstrates the segmentation and separation of objects from 2D-LM images. Here, we segmented nuclei, their boundaries, and detected interfaces between adjacent nuclei from a high-throughput screen on cultured cells [22] (Fig 2B, S2 Movie). The third and fourth examples demonstrate the segmentation of 3D EM and multicolor 3D LM datasets, where we segmented mitochondria from the mouse CA1 hippocampus [23] (Fig 2C, S3 Movie) and inner ear cell cytoplasm, nuclei and ribbon synapses from mouse inner ear cochlea [24] (Fig 2D, S4 Movie), respectively. In all cases, DeepMIB was able to achieve satisfactory results with modest time investment. Typically, the image segmentation is the slowest part of the imaging pipeline. Here, with DeepMIB, we were able to boost this part and decrease manual labor significantly, as the generation of ground truth was the only step requiring manual input. For example, in the fourth case, it took over 2 hours to manually and semi-automatically segment around 5% of the dataset that was used for training. In return, prediction of the whole dataset was completed in less than ten minutes, which was about 20 times faster compared to manual segmentation. The exact times required to train and apply the network for prediction of each four examples are given in S1 Supplementary Material. When necessary, the generated models can be further manually or semi-automatically proofread using MIB for quantitative analysis.

### Availability and future directions

The software is distributed either as an open-source Matlab code or as a compiled standalone application for Windows, MacOS, and Linux. DeepMIB is included into MIB distribution [15] (version 2.70 or newer), which is easy to install on any workstation or virtual machine preferably equipped with GPU. The Matlab version requires license for Matlab and Deep Learning, Computer Vision, Image Processing, Parallel Computing toolboxes. The compiled versions do





**Fig 2. Test datasets and acquired segmentation results.** (A) A middle slice of a serial section transmission electron microscopy dataset of the *Drosophila melanogaster* first instar larva ventral nerve cord supplemented with membrane prediction map and final segmentation of plasma membranes. (B) A random image of DNA channel from a high-throughput screen on human cultured osteocarcinoma U2OS cells (BBBC022 dataset, Broad BioImage Benchmark Collection) supplemented with prediction maps and final segmentation of nuclei, their boundaries (depicted in yellow),

and interfaces between adjacent nuclei (depicted in red). The inset highlights a cluster of three nuclei. (C) A slice from a focus ion beam scanning electron microscopy dataset of the CA1 hippocampus region supplemented with prediction maps and 3D visualization of segmented mitochondria. In the right image, the blue box depicts area used for training and green box for testing and evaluation of the network performance. (D) A maximum intensity projection of 3D LM dataset supplemented with predictions and segmentation of inner hair cell located in the cochlea of the mouse inner ear. The inner hair cell cytoplasm depicted in vermillion, their nuclei in dark blue, and ribbon synapses in yellow. Nuclei of the surrounding cells are depicted in light blue. The light blue box indicates the area used for training, dark blue box for validation and green box (magnified in the inset) for testing and evaluation. The dataset was segmented using 3D U-net Anisotropic architecture, which was specially designed for anisotropic datasets. Scale bars, (A) 200 nm, (C) 1  $\mu$ m, (D) 20  $\mu$ m. (Scale bar for (B) not known). All presented examples are supplemented with movies ([S1–S4 Movies](#)) and DeepMIB projects including datasets and trained network ([S1 Supplementary Material](#)).

<https://doi.org/10.1371/journal.pcbi.1008374.g002>

not require Matlab license and are easily installed using the provided installer. All distributions and installation instructions are available directly from MIB website (<http://mib.helsinki.fi>) or from GitHub (<https://github.com/Ajaxels/MIB2>). For a better understanding of the procedure and reducing the learning curve, DeepMIB has a detailed help section, online tutorials, and we provide workflows for all presented examples ([S1 Supplementary Material](#)). At the moment, DeepMIB offers four CNNs, but as a future perspective to fulfill the needs of more experienced users, we are aiming to increase the list, add new augmentation options and provide a configuration tool for designing of own networks or import networks trained elsewhere. As the software is distributed as an open-source code, it can be easily extended (for coding tutorials see MIB website) and future development on import/export of the generated networks would allow better interconnection with other deep learning software packages.

## Supporting information

**S1 Fig. Schematic representation of the 3D U-Net Anisotropic architecture.** The architecture is based on a standard 3D U-net, where the 3D convolutions and the Max Pooling layer of the 1<sup>st</sup> encoding level are replaced with the corresponding 2D operations (marked using “2D” label). To compensate, the similar swap is done for the last level of the decoding pathway. The scheme shows one of possible cases with patch size of 128x128x64x2 (height x width x depth x color channels), 2 depth levels, 32 first level filters and using “same” padding. This network architecture can be tweaked by modifying configurable parameters and it works best for anisotropic voxels with 1 x 1 x 2 (x, y, z) aspect ratio.

(TIF)

**S1 Supplementary Material. Details of installation details, online tutorials and example datasets.**

(DOCX)

**S1 Movie. Final segmentation results of the test dataset of serial section transmission electron microscopy of the *Drosophila melanogaster* first instar larva ventral nerve cord.** The video is accompanying [Fig 2A](#).

(MP4)

**S2 Movie. Final segmentation of nuclei (blue), their boundaries (yellow) and interfaces between adjacent nuclei (red) for random images from a high-throughput screen on human cultured osteocarcinoma U2OS cells (BBBC022 dataset, Broad BioImage Benchmark Collection).** The video is accompanying [Fig 2B](#).

(MP4)

**S3 Movie. Segmentation of mitochondria from the full focused ion beam scanning electron microscopy dataset of the CA1 hippocampus region.** The video is accompanying [Fig 2C](#).

(MP4)

**S4 Movie. A 3D LM dataset from mouse inner ear cochlea.** The shown dataset has two color channels: blue, CtBP2 staining of nuclei and ribbon synapses, and red, myosin 7a staining, highlighting inner and outer hair cells. The bottom slice, shown with the model represents the maximum intensity projection through the z-stack. The focus of the study was to segment the inner hair cells and their synapses thus the training and the validation sets were made around those cells omitting other cell type. The video is accompanying [Fig 2D](#). (MP4)

## Acknowledgments

We would like to thank Kuu Ikäheimo and Ulla Pirvola (the Auditory Physiology group, University of Helsinki) for kindly providing the inner ear dataset. The ISBI Challenge: Segmentation of neuronal structures in EM stacks [25] is acknowledged for evaluating the results of the 2D-EM segmentation. Dr. Konstantin Kogan (Institute of Biotechnology, University of Helsinki) is thanked for helping with testing of MacOS version. Prof. Jussi Tohka and Mr. Ali Abdollahzadeh (University of Eastern Finland), and Dr. Helena Vihinen (Institute of Biotechnology, University of Helsinki) are thanked for the valuable comments on the manuscript.

## Author Contributions

**Conceptualization:** Ilya Belevich, Eija Jokitalo.

**Data curation:** Ilya Belevich.

**Formal analysis:** Ilya Belevich.

**Funding acquisition:** Eija Jokitalo.

**Methodology:** Ilya Belevich.

**Project administration:** Eija Jokitalo.

**Resources:** Ilya Belevich, Eija Jokitalo.

**Software:** Ilya Belevich.

**Supervision:** Eija Jokitalo.

**Validation:** Ilya Belevich.

**Visualization:** Ilya Belevich.

**Writing – original draft:** Ilya Belevich.

**Writing – review & editing:** Ilya Belevich, Eija Jokitalo.

## References

1. Moen E, Bannon D, Kudo T, Graf W, Covert M, Van Valen D. Deep learning for cellular image analysis. *Nat Methods*. 2019; 16(12):1233–46. <https://doi.org/10.1038/s41592-019-0403-1> WOS:000499653100023. PMID: 31133758
2. Sullivan DP, Winsnes CF, Akesson L, Hjelmare M, Wiking M, Schutten R, et al. Deep learning is combined with massive-scale citizen science to improve large-scale image classification. *Nat Biotechnol*. 2018; 36(9):820–8. <https://doi.org/10.1038/nbt.4225> WOS:000443986000023. PMID: 30125267
3. Nitta N, Sugimura T, Isozaki A, Mikami H, Hiraki K, Sakuma S, et al. Intelligent Image-Activated Cell Sorting. *Cell*. 2018; 175(1):266–76. <https://doi.org/10.1016/j.cell.2018.08.028> WOS:000445120000029. PMID: 30166209
4. Weigert M, Schmidt U, Boothe T, Muller A, Dibrov A, Jain A, et al. Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nat Methods*. 2018; 15(12):1090–7. <https://doi.org/10.1038/s41592-018-0216-7> WOS:000451826200036. PMID: 30478326



5. Haberl MG, Churas C, Tindall L, Boassa D, Phan S, Bushong EA, et al. CDeep3M-Plug-and-Play cloud-based deep learning for image segmentation. *Nat Methods*. 2018; 15(9):677–80. <https://doi.org/10.1038/s41592-018-0106-z> WOS:000443439700021. PMID: 30171236
6. Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Lect Notes Comput Sc*. 2015; 9351:234–41. [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28) WOS:000365963800028.
7. Helmstaedter M, Briggman KL, Turaga SC, Jain V, Seung HS, Denk W. Connectomic reconstruction of the inner plexiform layer in the mouse retina. *Nature*. 2013; 500(7461):168–74. <https://doi.org/10.1038/nature12346> WOS:000322825500027. PMID: 23925239
8. Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O, editors. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2016*; 2016; Cham: Springer International Publishing.
9. Mathis A, Mamidanna P, Cury KM, Abe T, Murthy VN, Mathis MW, et al. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nat Neurosci*. 2018; 21(9):1281–9. Epub 2018/08/22. <https://doi.org/10.1038/s41593-018-0209-y> PMID: 30127430.
10. McQuin C, Goodman A, Chernyshev V, Kamensky L, Cimini BA, Karhohs KW, et al. CellProfiler 3.0: Next-generation image processing for biology. *PLoS Biol*. 2018; 16(7):e2005970. Epub 2018/07/04. <https://doi.org/10.1371/journal.pbio.2005970> PMID: 29969450; PubMed Central PMCID: PMC6029841.
11. Berg S, Kutra D, Kroeger T, Straehle CN, Kausler BX, Haubold C, et al. ilastik: interactive machine learning for (bio) image analysis. *Nat Methods*. 2019; 16(12):1226–32. <https://doi.org/10.1038/s41592-019-0582-9> WOS:000499653100022. PMID: 31570887
12. Gómez-de-Mariscal E, García-López-de-Haro C, Donati L, Unser M, Muñoz-Barrutia A, Sage D. DeepImageJ: A user-friendly plugin to run deep learning models in ImageJ. *bioRxiv*. 2019:799270. <https://doi.org/10.1101/799270>
13. Falk T, Mai D, Bensch R, Cicek O, Abdulkadir A, Marrakchi Y, et al. U-Net: deep learning for cell counting, detection, and morphometry (vol 16, pg 67, 2019). *Nat Methods*. 2019; 16(4):351–. <https://doi.org/10.1038/s41592-019-0356-4> WOS:000462620400032. PMID: 30804552
14. Urakubo H, Bullmann T, Kubota Y, Oba S, Ishii S. UNI-EM: An Environment for Deep Neural Network-Based Automated Segmentation of Neuronal Electron Microscopic Images. *Sci Rep-Uk*. 2019; 9:19413. <https://doi.org/10.1038/s41598-019-55431-0> WOS:000508836900007. PMID: 31857624
15. Belevich I, Joensuu M, Kumar D, Vihinen H, Jokitalo E. Microscopy Image Browser: A Platform for Segmentation and Analysis of Multidimensional Datasets. *Plos Biology*. 2016; 14(1):e1002340. <https://doi.org/10.1371/journal.pbio.1002340> WOS:000371882900009. PMID: 26727152
16. Arganda-Carreras I, Kaynig V, Rueden C, Eliceiri KW, Schindelin J, Cardona A, et al. Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification. *Bioinformatics*. 2017; 33(15):2424–6. Epub 2017/04/04. <https://doi.org/10.1093/bioinformatics/btx180> PMID: 28369169.
17. Piccinini F, Balassa T, Szkalitsky A, Molnar C, Paavola L, Kujala K, et al. Advanced Cell Classifier: User-Friendly Machine-Learning-Based Software for Discovering Phenotypes in High-Content Imaging Data. *Cell Syst*. 2017; 4(6):651–5 e5. Epub 2017/06/26. <https://doi.org/10.1016/j.cels.2017.05.012> PMID: 28647475
18. Badrinarayanan V, Kendall A, Cipolla R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans Pattern Anal Mach Intell*. 2017; 39(12):2481–95. Epub 2017/01/07. <https://doi.org/10.1109/TPAMI.2016.2644615> PMID: 28060704.
19. Linkert M, Rueden CT, Allan C, Burel JM, Moore W, Patterson A, et al. Metadata matters: access to image data in the real world. *J Cell Biol*. 2010; 189(5):777–82. Epub 2010/06/02. <https://doi.org/10.1083/jcb.201004104> PMID: 20513764; PubMed Central PMCID: PMC2878938.
20. Simard PY, Steinkraus D, Platt JC. Best practices for convolutional neural networks applied to visual document analysis. *Proc Int Conf Doc*. 2003:958–62. WOS:000185624000176.
21. Cardona A, Saalfeld S, Preibisch S, Schmid B, Cheng A, Pulokas J, et al. An integrated micro- and macroarchitectural analysis of the Drosophila brain by computer-assisted serial section electron microscopy. *PLoS Biol*. 2010; 8(10):e1000502. Epub 2010/10/20. <https://doi.org/10.1371/journal.pbio.1000502> PMID: 20957184; PubMed Central PMCID: PMC2950124.
22. Ljosa V, Sokolnicki KL, Carpenter AE. Annotated high-throughput microscopy image sets for validation. *Nat Methods*. 2012; 9(7):637. Epub 2012/06/30. <https://doi.org/10.1038/nmeth.2083> PMID: 22743765; PubMed Central PMCID: PMC3627348.
23. Knott GC M. FIB-SEM electron microscopy dataset of the CA1 hippocampus 2012. Available from: <https://www.epfl.ch/labs/cvlab/data/data-em/>.

24. Herranen A, Ikaheimo K, Lankinen T, Pakarinen E, Fritzsche B, Saarma M, et al. Deficiency of the ER-stress-regulator MANF triggers progressive outer hair cell death and hearing loss. *Cell Death Dis.* 2020; 11(2):100. Epub 2020/02/08. <https://doi.org/10.1038/s41419-020-2286-6> PMID: 32029702; PubMed Central PMCID: PMC7005028.
25. Arganda-Carreras I, Turaga SC, Berger DP, Ciresan D, Giusti A, Gambardella LM, et al. Crowdsourcing the creation of image segmentation algorithms for connectomics. *Front Neuroanat.* 2015; 9:142. <https://doi.org/10.3389/fnana.2015.00142> WOS:000365846500001. PMID: 26594156